Massachusetts Institute of Technology
C. S. Draper Laboratory
Cambridge, Massachusetts

LUMINARY Memo #138, Revision 1

To: Distribution
From: D. Eyles
Date: 6 March 1970
Subject: Variable Guidance Period Servicer

The final solution to powered flight TLOSS problems should be sought in cleaner logic than that which was devised to qualify P66 Auto for flight on Apollo 13. This solution only treats the symptom: by omitting occasional guidance computations it prevents jobs from piling up and being executed willy nilly, allowing the system to degrade gracefully not grossly. But there is no reason that the software must degrade at all. A Variable Guidance Period Servicer promises to let powered flight tolerate very high TLOSS with no perceptible ill effect, and it does so by simplifying rather than complicating. It attacks the core of the malignancy. By stretching the guidance period it in effect provides extra time: TGAIN.

Servicer is the navigation routines, a task and a job, which begin the job in which guidance is processed. Servicer gives its name to the whole job. Currently, Servicer is started punctually every 2 seconds, whether or not the previous pass has ended. It is proposed to let the period stretch and never start Servicer until its preceeding pass concludes.

A PCR to incorporate this solution in time for the Apollo 15 rope (sticking with a 14 very like 13) has been promised the next SCB -- providing it still seems feasible at MIT. This memo gives some reasons for putting in a variable Servicer, tells a little history, and describes the proposed Servicer structure.

Reasons

Reasons for having a variable guidance period fall under three main headings: safety, economy, flexibility.

Safety. It is now known that the stacking up of Servicer jobs which preceeds a 1201 or 1202 alarm in the event of TLOSS can cause completely wrong throttle and attitude commands, and in one special case, random branching. This happens because the Servicer job can clobber its own erasables. Luckily this did not happen to Apollo 11, which encountered TLOSS estimated at up to 15%. Perhaps cynically, no one I know is completely sure that time-snatching of that magnitude will never happen again. But even if we can be

completely sure about that, we cannot be sure that we will not soon be at the mercy of TLOSS caused by vibration. With TLOSS margins constantly shrinking due to accretions and real improvements like P66 Auto, we are entering a region where the "nominal" TLOSS (as yet unmeasured) may cause trouble. And the particular malignancy of the trouble is that it could cause the LM to manoeuvre and throttle wildly without warning (requiring astronaut intervention).. Excursions and alarums, not the other way around.

Economy. Incorporation of the variable Servicer now, at the cost of a somewhat more intense effort for Apollo 15, will save the enormous amount of time which would otherwise have to be spent in qualifying future ropes for flight in the presence of TLOSS and defining their TLOSS margins. An Apollo 15 rope can be designed to fly equally well at zero and at 33% TLOSS. Afterwards, unless drastic program changes are made, we will have great confidence in the ability of future ropes to put up with a TLOSS somewhere in between. Gone will be the necessity of laboriously TLOSS testing programs whenever a minor change is made. Also saved would be all the nervous energy expended in TLOSS controversies, sure to get less entertaining as time goes on and distracting from the more positive things to be done.

Flexibility. Without a variable guidance period capability we are at if not distinctly up against the wall so far as future additions to powered flight are concerned. I think it is naive to believe that beyond an "a priori" terrain model and delta-guidance there are no desirable goodies in store for the future. Without a variable Servicer they would have to be turned down no matter how good. A variable Servicer would also afford more flexibility in using programs. Presently, for instance, one would hesitate to call up a display with verb 16 during P66 or even P64. Verb 16 is said to take up nearly 10% of time when it is running. This means, among other things, that noun 92, which contains guidance thrust command in percent as a cue for the astronaut when he is throttling manually, may not be usable. (Further tests will show.)

History

A variable guidance period has been thought about at MIT at least since LUMINARY was split off from SUNDANCE -- and in abstract form for far longer. It was proposed in PCR 650 (November 1968) and in PCR 886 (August 1969) although apparently only the first reached an SCB. The reasons for it then included those given above, though of course the safety and economy considerations were rather less vivid than they are now. Mention was made in PCR 650 of the ease of nulling residuals with a guidance and display loop running at half a second, a concern which in the light of all that has happened since seems almost comical.

Around the time of PCR 650 Peter Adler and I created an underground version of LUMINARY called DIANA by MOONLIGHT as a test bed for powered flight improvements. A Variable Guidance Period Servicer was developed for DIANA. It was variable by command -- with extended verbs to expand or contract the loop time -- and had a granularity of 1/4 second. Thus it was structured differently from the present plan which adjusts to the time available without astronaut intervention. But despite the differences this experiment illuminates the magnitude and the nature of the task involved. In addition, some blocks of coding from DIANA's Servicer, for instance the Average-g computation, are suitable without modification for the new variable Servicer, which thus will benefit from tests already made. In DIANA all programs were modified to run with the variable Servicer and the P40s and the descent were tested successfully and repeatedly on both hybrid and digital simulators. I still have a few of these runs.

The current phase of the MIT Variable Guidance Period Servicer effort includes, besides the educational aspect, the making of a version of LUMINARY 145 with a variable Servicer built along the lines of current thinking. This program is called ZERLINA, and ought to be running soon.

Description

The outstanding feature of the proposed Servicer's structure is its simplicity. Servicer will no longer be an unholy coupling of the punctual PIPA task with an amorphous, laggard job. The READACCS task will be eliminated. The accelerometers will be read as part of the Servicer job

(under inhint of course) since with the guidance period (PGUIDE) computed rather than assumed their timing will be unimportant. There will be one and only one Servicer job always running. When it finishes it will go to the beginning and start over. Thus it cannot overlap itself. Servicer's restart protection becomes less unwieldy since READACCS no longer has to be protected separately. Since it is rot not reform that brings on maggots and silverfish, program reliability is enhanced. Running tranquilly in the foreground with a higher priority, as they do now, will be such off-line jobs as radar-reads, gyro-compensation, keystrokes, display jobs, some extended verbs (those that stay at PRIO 30), and monitor verbs. Once a cycle Servicer will go to sleep to allow lower priority extended verb activity.

Loop control logic, executed between the end of guidance (and display) and the next PIPA reading, will include: (1) Simple logic to insure a minimum Servicer period. Except perhaps for the P40s, which could run much faster if there were any reason to, 2 seconds would be a sensible minimum. This means that with the present guidance programs and low TLOSS the new Servicer will run at 2 seconds like the present one. Also, a minimum period of 2 seconds will prevent the loss of any once-every-2-second downlink data. (2) An alarm in the event that the Servicer period exceeds some limit (say 4 seconds) because for scaling purposes some upper bound needs to be enforced and because if there were ever a really monstrous TLOSS we would like at least to know about it.

The rationale for the design of the new Servicer is this: If no item in the loop is time-critical and the loop cannot overlap itself, then the loop is not sensitive to TLOSS.

We pay for this immunity with a little more algebra in Average-g and the guidance equations. One item will be added and one replaced in the Servicer-Guidance interface. A double-precision erasable PGUIDE, copied at COPYCYCL from PGUIDE1 computed at the time of the PIPA reading, will contain guidance period in units of 2^{14} centiseconds. And vector G, an acceleration scaled in units of 2^{-6} m/cs^2, will replace GDT/2, often used as an acceleration (under assumptions about the guidance period)

but really one second's worth of delta-v due to gravity. For programs outside of Servicer the changes necessary usually have to do with the handling of the gravity vector. GDT/2 was originally contrived to simplify the Average-g equation -- G will better suit the guidance equations. Minor changes will have to be made in THROTTLE, where ABDELV is now used as an acceleration, and in FINDCDUW to prevent overshoot in case more than 2 seconds elapse between calls. Since the velocity and altitude radar reads would remain joined but could no longer be synchronized with READACCS, an extrapolation of Average-g data to the time of the radar read will be required, but the equation is a very familiar one. 1/PIPADT for the PIPA compensation routine will have to be computed and its scaling changed to accomodate a PGUIDE longer than 2.55 seconds, but this is easily done. Landing analog displays require trivial changes. Before calling the display routines the Servicer job will in general raise its priority temporarily in order to give the off-line display jobs that result in a higher priority than Servicer's traditional PRIO 20. Instead of going to ENDOFJOB at the end of the Servicer job, guidance programs would transfer control to PIPCYCLE instead to continue the loop; it will have to be insured that every branch eventually leads to PIPCYCLE.

Thus in almost random order are the changes that appear to be involved in implementing the Variable Guidance Period Servicer. The main purpose of this memo is to elicit comments and to uncover any strong objections to the proposed plan. Later memos will describe the work done on ZERLINA: the details of the various changes, and of the off-line testing.